Experiments in Modern Physics

# Introductory Lab: PC-DAQ

Rex Tayloe
Department of Physics, Indiana University

1/1/03
Last Revised: 9/4/03

**Goal**

Become familiar with operating and writing code to address the National Instruments DAQPad-6020E multifunction I/O device.

**Introduction**

In this lab you will get acquainted with the National Instruments [1] DAQPad-6020E multifunction I/O device (the "DAQPad") [2]. After this introduction, you will then be able to use this device to acquire data for other experiments. You will also be introduced to the Microsoft Visual C++ programming environment that is needed to program and readout the device.

You will do these tasks:
1. Startup the DAQPad and test it with the provided test software.
2. Create and run a program with Visual C++ (starting with an example provided).
3. Run an existing program to acquire data, and then analyze this data to characterize a battery.
4. Write your own program to investigate a feature of the DAQPad.

**Equipment**

- Computer with NI-DAQ software ("MPL3")
- DAQPad-6020E
- voltmeter, (old) battery, variable resistance, misc. cables.
- DAQPad manuals (Paper and/or on the computer)

**Preparation**

- Read the Section in Ref [3] about ADCs.
- Read the introductory part of the the DAQPad manual[2], available as hardcopy near the computer, on the computer as a .pdf file, or at the National Instruments www site.
- Create a voltage source by putting a (sufficiently large) resistance across the battery. You will use this to exercise the DAQPad.

**Task 1: Startup and test the DAQPad**

- Start and/or login to the MPL3 computer.
- Make sure the DAQPad power switch is off, plug in the power supply and USB cable.
- Turn on the DAQPad. You should hear MPL3 acknowledge the connection.
- Double-click on the Measurement & Automation desktop icon (or find from program menu).

- Find the DAQPad under "devices and interfaces" and run the test panel. Plug in a voltage source to an input channel and read it. Hook a voltmeter up to an output channel and verify operation.
- Read the DAQPad manual to understand the analog input and output channels and the various modes of operation (differential/single-ended)

**Task 2: Create and run the AIonePoint program with Visual C++**
- Following the instructions in the Appendix, setup a new project under the C:\MPLDAQ area using MS Visual C++.
- Copy the AIonePoint.C code from the NI-DAQ area.
- Run it with your voltage source input to verify that it is reading correctly. Change the code to read a different channel, with a different gain, etc.

**Task 3: Run the DaqToDisk program to collect data and characterize a battery**
- In MPLDAQ there is a project called “daqtodisk”. Compile, link, and run this program.
- Note that (as currently written) you have to edit the C code to change the channel, gain, sample rate, etc
- Hook up your voltage source and record the voltage as the battery discharges. It is useful to look at an older battery that discharges faster.
- Plot and analyze this data to characterize the voltage source (the battery). Use the ideal source-internal resistance model to characterize the battery. How does the internal resistance change with time? How does the power through the external resistance change with time? What is the total energy delivered? How long with the battery last?

**Task 4: Create your own program to investigate a feature of the DAQPad**
- Starting with the C++ project that you set up in Task 2, edit the program and investigate another feature of the DAQPad.

**References**

[1] http://www.ni.com
[2] DAQPad-6020E User Manual, National Instruments, Sept. 1998.
[3] W.R. Leo, Techniques for Nuclear and Particle Physics Experiments, Springer, 1987.

## Appendix

**Miscellaneous Notes**
- The DAQ area on the MPL3 computer is C:\MPLDAQ
- In the following, the NI-DAQ directories are at (and under): C:\Program Files\National Instruments\NI-DAQ. There is a shortcut under the MPLDAQ directory as well.
- General programming help is in (pdf file) “NI-DAQ for PC Version 6.9.1 User Manual” under Programs>>National Instruments>>NI-DAQ>>Documents.
- The DAQ function reference help is under Programs>>National Instruments>>NI-DAQ>>NI-DAQ Help

- Be careful not to change any of the original files in the NI-DAQ directories. Always make copies. Understand how this works with the Visual C++ environment.
- When running with Visual C++ the default mode is DEBUG, so sometimes you will find the executable programs in the Debug directory.

### How to create a DAQPad program (setup a new project in C++) from a NI example

- following instructions from www.ni.com document: "**How to Compile and Run NI-DAQ Examples with Microsoft Visual C++**" (located on desktop of MPL3 computer).
- Open MS Visual C++
- Then:

1. Start by opening a new project by going to **File » New**.

2. The Projects tab should come up by default. You will be asked for the name and type of project. Select the "Win32 Console Application" as the type and give your project a name.

3. Select "Empty Project" in the next step. Once you have a new project, it will show your folder menu on the left and the file screen on the right. When you double click on any file you have in your project it will be displayed on the right side. To make navigation easier, change from "Class View" to "File View".

4. To integrate the source code for most of the data acquisition application examples, you just need to add two libraries and header files into your project. Go to the **Project » Add to Project » File** menu.

5. Select the **NI-DAQ\LIB** directory and add the **nidaq32.lib** file and the **nidex32.lib** file. You should also insert the **nidaqex.h** and **nidaq.h** file from the **NI-DAQ\Include** folder. **

6. Go to **Tools » Options** and select the Directories tab.

7. Set the option to be "Include Files". Make sure that you point to the same directory in **NI-DAQ\INCLUDE**. (This may be set already for you).

8. From the same tab, set the option to be "Libraries" and set a path for **NI-DAQ\LIB**. (This may be set already for you).

9. Now add to your project the source code and header (if needed) that you want to use. To do this, copy an example from the NI-DAQ examples directory (C:\Program Files\National Instruments\NI-DAQ\Examples\VisualC\Ai) into the project directory (rename if you wish). Add this C file to the project by the **Project » Add to Project » File** steps as above.

10. To build: Change to "File View", left-click on the C program you just added, **Build » Build <progname> .exe** to compile and link, then , **Build » Execute <progname> .exe** to run. This can also be run from outside the C++ environment (double-click on the .exe file).